

Requested Patent JP10187475A

Title:
METHOD FOR GENERATING INSTRUCTION SEQUENCE FOR TEST FOR INFORMATION PROCESSOR ;

Abstracted Patent JP10187475 ;

Publication Date: 1998-07-21 ;

Inventor(s): ICHIKAWA FUMIO ;

Applicant(s): FUJITSU LTD ;

Application Number: JP19960346335 19961225 ;

Priority Number(s): ;

IPC Classification: G06F11/22 ;

Equivalents:

ABSTRACT:

PROBLEM TO BE SOLVED: To generate an instruction sequence for test with which back branching instructions over wide range can be integrated without becoming an infinite loop, the verification of branch close to real operation due to various flow control sentences used for an operation system or the like is sufficiently performed and a branch predictive test can be performed while effectively utilizing the characteristics of random instruction generation.

SOLUTION: At the arbitrary timing of process for forward generating a random instruction sequence, an empty area NR is secured for the prescribed number of instructions and just before the empty area NR, a non-conditional forward branching instruction UFB defining the branch destination as just after the empty area NR is generated. At the arbitrary timing of following random instruction sequence generating process, a non-conditional back branching instruction UBB defining the branch destination as the empty area NR is generated and a non-conditional forward branching instruction UFB defining the part after the non-conditional back branching instruction UBB at the branch source as the branch destination or a conditional forward branching instruction CFB is inserted into the empty area NR so that the instruction sequence for test can be generated.

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-187475

(43)公開日 平成10年(1998)7月21日

(51)Int.Cl.
G 0 6 F 11/22

識別記号
3 1 0

F I
G 0 6 F 11/22
3 1 0 C
3 1 0 U

審査請求 未請求 請求項の数8 OL (全 10 頁)

(21)出願番号

特願平8-346335

(22)出願日

平成8年(1996)12月25日

(71)出願人 000005223

富士通株式会社
神奈川県川崎市中原区上小田中4丁目1番
1号

(72)発明者 市川 文男

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

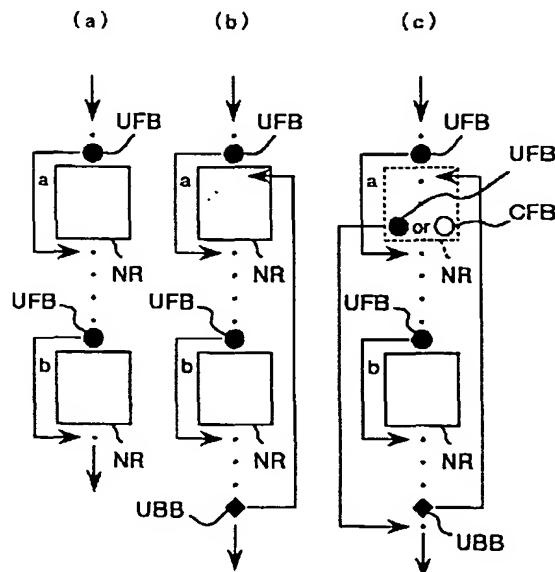
(74)代理人 弁理士 井桁 貞一

(54)【発明の名称】 情報処理装置の試験用命令列生成方法

(57)【要約】

【課題】 無限ループになることなく、広範囲に亘るバック分岐命令を組み込むことができ、オペレーション・システム等で使用される各種フロー制御文による実際の運用に近い分岐の検証を充分に行い、ランダム命令生成の特質を活かした分岐予測試験を行うことができる試験用命令列を生成すること。

【解決手段】 フォワード方向にランダム命令列を生成する過程の任意のタイミングで、所定数の命令分の空き領域NRを確保し、該空き領域NRの直前に分岐先を空き領域NRの直後とした無条件フォワード分岐命令UFBを生成し、これ以降のランダム命令列生成過程の任意のタイミングで、分岐先を空き領域NRにした無条件バック分岐命令UBBを生成し、空き領域NRには分岐元の無条件バック分岐命令UBBより以降を分岐先とした無条件フォワード分岐命令UFBあるいは条件付フォワード分岐命令CFBを挿入して試験用命令列を生成する。



【特許請求の範囲】

【請求項1】 フォワード方向にランダム命令列を生成する過程の任意のタイミングで、所定数の命令分の空き領域を確保すると共に、空き領域の直前に、分岐先を空き領域の直後とした無条件フォワード分岐命令を生成し、これ以降のランダム命令列の生成過程の任意のタイミングで、分岐先を前記空き領域にしたバック分岐命令を生成し、空き領域には分岐元のバック分岐命令より以降を分岐先としたフォワード分岐命令を挿入することを特徴とする情報処理装置の試験用命令列生成方法。

【請求項2】 一つのランダム命令列において空き領域を複数箇所に確保し、バック分岐命令の分岐先をその複数の空き領域より選択してバック分岐幅を選択設定することを特徴とする請求項1に記載の情報処理装置の試験用命令列生成方法。

【請求項3】 一つの空き領域を複数の命令分の空き領域として設定し、空き領域にはフォワード分岐命令以外にランダム命令を挿入することを特徴とする請求項1または2に記載の情報処理装置の試験用命令列生成方法。

【請求項4】 空き領域に挿入するフォワード分岐命令は、無条件分岐命令であることを特徴とする請求項1～3のいずれか一つに記載の情報処理装置の試験用命令列生成方法。

【請求項5】 空き領域に挿入するフォワード分岐命令は、プログラムの実行に関係なく変化する制御情報を分岐条件値とした条件分岐命令であることを特徴とする請求項1～3のいずれか一つに記載の情報処理装置の試験用命令列生成方法。

【請求項6】 フォワード方向にランダム命令列を生成する過程の任意のタイミングで、プログラムの実行に関係なく変化する制御情報値を分岐条件値とした条件付きのバック分岐命令を生成することを特徴とする情報処理装置の試験用命令列生成方法。

【請求項7】 前記制御情報値は複数ビットの情報値であり、そのビット数の選択あるいは分岐条件の判定を逆にすることにより繰り返しループ回数を可変設定することを特徴とする請求項5または6に記載の情報処理装置の試験用命令列生成方法。

【請求項8】 請求項4、5、6に記載の試験用命令列生成方法により生成される命令列のうち、少なくとも二種類の命令列を組み合わせて一つの試験用命令列を生成することを特徴とする情報処理装置の試験用命令列生成方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、情報処理装置におけるプロセッサの機能を検証するための試験用命令列を生成する方法に関し、特に、オペレーション・システムやアプリケーション・プログラムで使用される命令のランダムな組み合わせによる試験用命令列をフォワード

方向に生成する情報処理装置の試験用命令列生成方法に関する。

【0002】

【従来の技術】計算機システムなどの情報処理装置におけるプロセッサの機能を試験用命令列による試験運転によって検証することは既に行われている。この試験に使用される命令列の生成は、乱数を用いてランダムに命令を選択してフォワード方向にランダム命令列を生成するのが一般的である。

【0003】図13は、従来における試験用命令列の生成手順を示している。この試験用命令列生成手順では、先ず、図14(a)に示されているように、命令生成域 I GR (Instruction-Generated Region) にランダム・データを作成し(ステップS10)、命令生成域 I GR の先頭アドレス HA (Head Address) をポインタに設定する(ステップS20)。

【0004】つぎに、ポインタが指すランダム・データを利用して命令種別(单一命令、複合命令)を判別抽出する(ステップS30)。

【0005】判別抽出の結果、单一命令であれば、ポインタが指すランダム・データを利用して図14(b)に示されているような命令テーブル IT₁ (Instruction Table)から命令(单一命令)を抽出し(ステップS40)、ポインタが指す領域に命令を生成し(ステップS50)、ポインタをフォワード方向に更新する(ステップS60)。ここで、生成される命令には、load, store, add, sub, 分岐などがあり、分岐命令に関しては、暴走状態や無限ループにならないことを保証するために、分岐先をフォワード方向に設定する制約があり、分岐命令から分岐先までのアドレスには空き領域を生じさせることなく任意の命令が組み込まれる。

【0006】これに対し、複合命令であれば、ポインタが指すランダム・データを利用して図14(c)に示されているような命令テーブル IT₂ から命令列(ルーチン)を抽出し(ステップS70)、ポインタの指す領域に命令を生成し(ステップS80)、ポインタをフォワード方向に更新する(ステップS60)。このルーチン中の分岐命令は、無限ループにならないことを保証するために、分岐先をルーチン中に限ってバック方向に設定することが可能になっている。

【0007】上述のステップS30～S80は、ポインタが命令生成域 I GR の最終アドレスを指す(ステップS90)まで繰り返され、これによって試験用の一つのランダム命令列がフォワード方向に生成される。

【0008】

【発明が解決しようとする課題】上述した従来における試験用命令列生成方法では、試験用のランダム命令列はアドレスの昇順に命令が組み立てられるから、分岐命令は、暴走状態や無限ループにならないことを保証するために、アドレスの昇順にしか分岐できないフォワード分

岐命令に制限され、オペレーション・システムやアプリケーション・プログラムで使用されるGOTO文やFOR文やDO文などのフロー制御文による実際の運用に近い分岐の検証を行うことについて、充分な試験用命令列を生成することができない。

【0009】この問題を解決するために、図14(c)に示されているように、バック系の分岐命令を含んだルーチンを予め準備しておき、試験用命令列の作成過程で、そのルーチンを複合命令として組み込むことが行われているが、これでは、分岐先がルーチン中に限られるから、全体の命令列から見れば、極小範囲でしかバック分岐を実現することができず、また、予測分岐機能について充分な検証試験を行うことができず、さらに、ランダム命令生成と云う特質を活かすこともできない。

【0010】この発明は、上述のような問題点を解決するためになされたものであり、ランダムに生成可能な命令の一つとして、無限ループになることなく、広範囲に亘るバック分岐命令を組み込むことができ、オペレーション・システムやアプリケーション・プログラムで使用されるGOTO文やFOR文やDO文などのフロー制御文による実際の運用に近い分岐の検証を充分に行い、ランダム命令生成の特質を活かした予測分岐試験(バック系、ロシング分岐、ネスト)を行うことができる試験用命令列を生成する情報処理装置の試験用命令列生成方法を提供することを目的としている。

【0011】

【課題を解決するための手段】上述の目的を達成するために、請求項1に記載の発明は、フォワード方向にランダム命令列を生成する過程の任意のタイミングで、所定数の命令分の空き領域を確保すると共に、空き領域の直前に、分岐先を空き領域の直後とした無条件フォワード分岐命令を生成し、これ以後のランダム命令列の生成過程の任意のタイミングで、分岐先を前記空き領域にしたバック分岐命令を生成し、空き領域には分岐元のバック分岐命令より以降を分岐先としたフォワード分岐命令を挿入する情報処理装置の試験用命令列生成方法である。

【0012】請求項1に記載の情報処理装置の試験用命令列生成方法では、ランダムに生成可能な命令の一つとして、無限ループの問題を生じることなくバック分岐命令がランダム命令列に組み込まれ、GOTO文やFOR文などのフロー制御文による実際の運用に近い分岐の検証試験が行われ得るようになる。

【0013】請求項2に記載の発明は、一つのランダム命令列において空き領域を複数箇所に確保し、バック分岐命令の分岐先をその複数の空き領域より選択してバック分岐幅を選択設定する請求項1に記載の情報処理装置の試験用命令列生成方法である。

【0014】請求項2に記載の情報処理装置の試験用命令列生成方法では、分岐幅を可変設定できる。

【0015】請求項3に記載の発明は、一つの空き領域

を複数の命令分の空き領域として設定し、空き領域にはフォワード分岐命令以外にランダム命令を挿入する請求項1または2に記載の情報処理装置の試験用命令列生成方法である。

【0016】請求項3に記載の情報処理装置の試験用命令列生成方法では、空き領域にフォワード分岐命令以外にランダム命令が挿入されることで、バイアス処理のタイミングが変化し、クリティカルな分岐テストが行われ得るようになる。

【0017】請求項4に記載の発明は、空き領域に挿入するフォワード分岐命令が、無条件分岐命令である請求項1～3のいずれか一つに記載の情報処理装置の試験用命令列生成方法である。

【0018】請求項4に記載の情報処理装置の試験用命令列生成方法では、ランダム分岐命令(GOTO文)が自動生成される。

【0019】請求項5に記載の発明は、空き領域に挿入するフォワード分岐命令が、プログラムの実行に関係なく変化する制御情報値を分岐条件値とした条件分岐命令である請求項1～3のいずれか一つに記載の情報処理装置の試験用命令列生成方法である。

【0020】請求項5に記載の情報処理装置の試験用命令列生成方法では、FOR文命令列あるいはFOR文ネスト命令列が自動生成される。

【0021】請求項6に記載の発明は、フォワード方向にランダム命令列を生成する過程の任意のタイミングで、プログラムの実行に関係なく変化する制御情報値を分岐条件値とした条件付きのバック分岐命令を生成する情報処理装置の試験用命令列生成方法である。

【0022】請求項6に記載の情報処理装置の試験用命令列生成方法では、DO文命令列あるいはDO文ネスト命令列が自動生成される。

【0023】請求項7に記載の発明は、前記制御情報値は複数ビットの情報値であり、そのビット数の選択あるいは分岐条件の判定を逆にすることにより繰り返しループ回数を可変設定する請求項5または6に記載の情報処理装置の試験用命令列生成方法である。

【0024】請求項7に記載の情報処理装置の試験用命令列生成方法では、FOR文命令列、FOR文ネスト命令列、DO文命令列、DO文ネスト命令列における繰り返しループ回数が可変設定される。

【0025】請求項8に記載の発明は、請求項4、5、6に記載の試験用命令列生成方法により生成される命令列のうちの、少なくとも二種類の命令列を組み合わせて一つの試験用命令列を生成する情報処理装置の試験用命令列生成方法である。

【0026】請求項8に記載の情報処理装置の試験用命令列生成方法では、GOTO文、FOR文、DO文が混在した試験用命令列が生成される。

【0027】

【発明の実施の形態】以下に添付の図を参照してこの発明に係る情報処理装置の試験用命令列生成方法の実施の形態を詳細に説明する。

【0028】なお、以下の説明において、空き領域NR(Null Region)を□印、無条件フォワード分岐命令UFB(Unconditional Forward Branch instruction)を●印、条件付フォワード分岐命令CFB(Conditional Forward Branch instruction)を○印、無条件バック分岐命令UBB(Unconditional Back Branch instruction)を◆印、条件付バック分岐命令CBB(Conditional Back Branch instruction)を△印、分岐先候補のポインタP(Pointer)を★印でそれぞれ示すことがある。なお、各図において、破線で示す空き領域NRは、バック分岐命令実行後の状態を示している。

【0029】(実施の形態1)この発明による情報処理装置の試験用命令列生成方法の実施の形態1の概要について、図1、図2を参照して説明する。

【0030】(試験用命令列生成手順)まず、試験用命令列は、以下の手順で生成する。

【0031】(1)図1(a)に示されているように、フォワード方向にランダム命令列を生成する過程の任意のタイミングで、所定数の命令分の空き領域NR(□)を確保する。この空き領域NR(□)の直前には、分岐先を空き領域NR(□)の直後とした無条件フォワード分岐命令UFB(●)を生成する。

【0032】フォワード分岐命令UFB(●)が設けられることにより、命令列の途中に空き領域NR(□)が存在しても、命令の実行に影響を与えることがない。換言すれば、生成されたランダム命令列は、空き領域NR(□)に影響なく正常に動作することを保証される。

【0033】(2)空き領域NR(□)が作成されれば、空き領域NR(□)の有効空間数と、各空き領域NR(□)ごとに、空き領域NR(□)の先頭ポインタ(a, b, c, d...)と空き領域NR(□)に入る命令数Nを、図2(a)に示されているような、空き領域ポインタ・テーブルPT(Pointer Table)に登録する。

【0034】この空き領域ポインタ・テーブルPTは、無条件バック分岐命令UBB(◆)を生成するときのバック分岐の分岐先を決定するために使用され、分岐先の選択設定に際して、作成されたいくつかの空き領域NR(□)から任意の一つを選択することにより、分岐幅を可変設定できる。

【0035】(3)その後のランダム命令列の生成過程の任意の時点で、図1(b)に示されているように、無条件バック分岐命令UBB(◆)を生成する。無条件バック分岐命令UBB(◆)の分岐先は、上記(2)で作成された空き領域ポインタ・テーブルPTから任意の空き領域NR(□)を取り出し、その空き領域NR(□)の先頭とする。取り出された空き領域NR(□)のデータ

(先頭ポインタaと命令数N)は、図2(b)に示されているように、空き領域ポインタ・テーブルPTより削除する。

【0036】(4)選択された空き領域NR(□)には、図1(c)に示されているように、先頭ポインタより命令数N-1個以内のランダム命令(分岐命令を除く)を順次生成し、最後に無条件フォワード分岐命令UFB(●)あるいは条件付フォワード分岐命令CFB(○)を生成する。この無条件フォワード分岐命令UFB(●)の分岐先は無条件バック分岐命令UBB(◆)の直後とする。条件付フォワード分岐命令CFB(○)の分岐条件値は、プログラムの実行に関係なく変化する値、例えば、タイマ値を分岐条件値とする。

【0037】これらの処理により、ランダムに生成可能な命令の一つとして、無条件バック分岐命令UBB(◆)をランダム命令列に組み込むことが可能になり、無限ループとなる問題も解消される。

【0038】上述のような無条件バック分岐命令UBB(◆)を複数生成することにより、GOTO文の組み合わせでロング分岐が可能になる。

【0039】また、無条件バック分岐命令UBB(◆)で分岐した分岐先の空き領域NR(□)に条件付フォワード分岐命令CFB(○)を組み込むことで、FOR文命令列あるいはFOR文ネスト命令列によるフロー制御の命令列を生成することができる。

【0040】このフォワード分岐命令CFB(○)の分岐条件値をタイマ等の常時加算あるいは常時減算される制御情報値とし、その下位nビットが0であるか否かにより分岐させることにより、繰り返しループ回数を可変設定することができ、また、条件分岐の判定を逆にすることによっても繰り返しループ回数を変化させることができる。

【0041】(GOTO文によるフロー制御命令列生成手順)つぎに、バック分岐命令の組み合わせによってGOTO文によるフロー制御の命令列を生成する具体的な手順を説明する。この場合、バック分岐命令を組み合わせることにより、ランダムな分岐が可能になる。

【0042】(1)最初に、バック分岐命令の個数を決定する。この個数が多いほど、分岐先の距離を長くできる。

【0043】(2)命令生成過程で、無作為に無条件フォワード分岐命令UFB(●)、または無条件バック分岐命令UBB(◆)を選択し、そのそれにおいて以下の処理を実行する。

【0044】(2-1)無条件フォワード分岐命令UFB()を選択した場合:図3(a)に示されているように定義した命令列を展開する。ここで、フォワード分岐命令UFB()は、無条件分岐命令あるいは条件設定命令と条件付分岐命令とを組み合わせて必ず分岐するよう考慮された命令列であり、無条件フォワード分岐命

令UFB()は、フォワード方向に分岐先を有し、フォワード分岐命令UFB()と分岐先との間に数命令分の空き領域NR(□)を確保する。

【0045】これにより、所定数の命令分の空き領域NR(□)を確保し、その空き領域NR(□)の直前に、分岐先を空き領域の直後としたフォワード分岐命令UFB(●)を生成したことになる。

【0046】一つの空き領域NR(□)を確保する度に、図4(a), (b)に示されているように、空き領域NR(□)の有効空間数を更新し、空き領域NR(□)の先頭ポインタと命令数を空き領域ポインタ・テーブルPTに登録する。

【0047】(2-2)無条件バック分岐命令UBB(◆)を選択した場合：図3(b)に示されているように定義した命令列を展開する。バック分岐の分岐先は空き領域ポインタ・テーブルPTに登録されている空き領域の中から任意に設定する。

【0048】空き領域NR(□)には分岐命令以外の任意の命令を埋め込み、空き領域NR(□)の最後にはフォワード分岐命令(無条件フォワード分岐命令相当)UFB(●)を埋め込む。この無条件フォワード分岐命令UFB(●)は分岐してきた無条件バック分岐命令UBB(◆)より以降を分岐先としている。この分岐先は分岐元の無条件バック分岐命令UBB(◆)の直後、あるいは分岐元のバック分岐命令より数命令分、進んだ位置とする。図3(b)は後者を示しており、無条件バック分岐命令UBB(◆)の生成に併せて空き領域NR(□)が確保される。これにより空き領域NR(□)の個数を多く取れる。

【0049】バック分岐が設定されれば、図4(b)に示されているように、使用した空き領域NR(□)のデータを空き領域ポインタ・テーブルPTから削除し、新たに生成した空き領域NR(□)のデータを空き領域ポインタ・テーブルPTに登録する。

【0050】無条件バック分岐命令UBB(◆)が指定個数、例えば、4個になるまで繰り返す。

【0051】なお、無条件バック分岐命令UBB(◆)を選択したときに、有効な空き領域NR(□)がない場合には、フォワード分岐に切り換える。

【0052】(3)これにより、図5に示されているように、フォワード方向にランダム命令列を生成する過程の任意のタイミングで、所定数の命令分の空き領域NR(□)を確保し、空き領域NR(□)の直前に、分岐先を空き領域NR(□)の直後とした無条件フォワード分岐命令UFB(●)を生成し、これ以降のランダム命令列の生成過程の任意のタイミングで、分岐先を空き領域NR(□)にした無条件バック分岐命令UBB(◆)を生成し、空き領域NR(□)には無条件バック分岐命令UBB(◆)より以降を分岐先とした無条件フォワード分岐命令UFB()を挿入することが行われ、ラング

分岐のランダム分岐命令(GOTO文)を含む試験用命令列が出来上がる。

【0053】これにより、ロング分岐のランダム分岐命令(GOTO文)の組み合わせによる予測分岐試験が行われ得るようになる。

【0054】また、無条件バック分岐命令UBB(◆)後のつぎの分岐命令までに、換言すれば、空き領域NR(□)に、分岐命令以外のランダム命令が組み込まれることによってバイオライン処理のタイミングが変化し、クリティカルな分岐テストが行われ得るようになる。

【0055】なお、図4(a)は図5の時点 t_1 における空き領域ポインタ・テーブルPTの状態を、図4(b)は図5の時点 t_2 における空き領域ポインタ・テーブルPTの状態をそれぞれ示している。なお、図5における①～④の数値は、バック分岐の実行順序を示したものである。

【0056】(FOR文ネスト命令列生成手順)つぎに、FOR文ネストの例について説明する。バック分岐命令で分岐した空き領域に条件分岐命令を組み込みことで、FOR文によるフロー制御の命令列の生成が可能になる。また、これを組み合わせることにより、FOR文ネスト命令列を生成できる。

【0057】(1)最初に、ネスト数を決定する。これはバック分岐命令の個数と等価である。

【0058】(2)命令列生成過程で、図6(a)に示されているように、ネスト数分の無条件フォワード分岐命令UFB(●)を無作為に選択し、ネスト数分、ここでは3個の空き領域NR(□)を確保する。無条件フォワード分岐命令UFB(●)では、図3(a)に示されているように定義した命令列を同様に展開する。

【0059】(3)ネスト数分の空き領域NR(□)が確保された以降の命令列生成過程の任意の時点で、図6(b)に示されているように、ネスト数分の無条件バック分岐命令UBB(◆)を選択する。無条件バック分岐命令UBB(◆)が選択されれば、図7に示されているように定義した命令列を展開する。

【0060】各バック分岐命令UBB(◆)の分岐先の設定に際しては、図2に示されているような空き領域ポインタ・テーブルPTを参照してネストの型になるように空き領域NR(□)を選び、選択された空き領域NR(□)の先頭とする。分岐先に選択された空き領域NR(□)のデータは空き領域ポインタより消去されるから、分岐先は重複することなく設定される。

【0061】(4)選択された各空き領域NR(□)には分岐命令以外の任意の命令を埋め込み、空き領域NR(□)の最後には条件付フォワード分岐命令CFB(○)を埋め込む。この条件付フォワード分岐命令CFB(○)の分岐先は分岐元の無条件バック分岐命令UBB(◆)の直後とする。

【0062】条件付フォワード分岐命令CFB(○)の

分岐条件値はタイマ受信命令により得られるカウンタ（タイマ）の値であり、そのカウンタの下位nビットの値が0であれば、分岐先へ分岐する。これにより繰り返しループ回数が設定される。図8に示されているように、下位ビットn=1の場合には、ループ確率、非ループ確率ともに1/2になり、下位ビットn=2の場合には、ループ確率は3/4、非ループ確率1/4になり、下位ビットn=3の場合には、ループ確率は7/8、非ループ確率1/8になるから、下位ビットnの選択により繰り返しループ回数を変化させ、ネストから抜け出す速さを調節することができる。

【0063】また、カウンタの下位nビットの値が0でなければ、分岐先へ分岐することもでき、この場合も、同様に下位ビットnの選択により、図8に示されているように、ループ確率、非ループ確率を変化させてネストから抜け出す速さを調節することができる。

【0064】(5)これにより、図6(c)に示されているように、フォワード方向にランダム命令列を生成する過程の任意のタイミングで、所定数の命令分の空き領域NR(□)を確保し、空き領域NR(□)の直前に、分岐先を空き領域NR(□)の直後とした無条件フォワード分岐命令UFB(●)を生成し、これ以降のランダム命令列の生成過程の任意のタイミングで、分岐先を空き領域NR(□)にした無条件バック分岐命令UBB(◆)を生成し、空き領域NR(□)には無条件バック分岐命令UBB(◆)より以降を分岐先とした条件付フォワード分岐命令CFB(○)を挿入することが行われ、FOR文のネストを含む試験用命令列が出来上がる。

【0065】これにより、FOR文命令列およびFOR文ネスト命令列による予測分岐試験が行われ得るようになる。

【0066】(実施の形態2)この発明による情報処理装置の試験用命令列生成方法の実施の形態2について、図9～図11を参照して説明する。実施の形態2では、フォワード方向にランダム命令列を生成する過程の任意のタイミングで、プログラムの実行に関係なく変化する制御情報値を分岐条件値とした条件付バック分岐命令CBB(◇)を生成する。

【0067】以下に、試験用命令列の生成手順を説明する。

【0068】(1)最初に、ネスト数を決定する。

【0069】(2)命令生成過程で、図9(a)に示されているように、ネスト数分の分岐先候補のポインタP(★)を無作為に抽出し、図10に示されているようなポインタ・テーブルPTに登録する。

【0070】(3)ネスト数分のポインタP(★)が設定されれば、それ以降の命令列生成過程の任意の時点で、図9(b)に示されているように、ネスト数分の条件付バック分岐命令CBB(◇)を選択する。条件付バ

ック分岐命令CBB(◇)の分岐先はポインタ・テーブルPTに登録されているポインタP(★)によるネスト位置より選択設定される。

【0071】条件付バック分岐命令CBB(◇)の分岐条件値はタイマ受信命令により得られるカウンタ（タイマ）の値であり、そのカウンタの下位nビットの値が0であれば、分岐先へ分岐する。これにより繰り返しループ回数が設定される。図11に示されているように、下位ビットn=1の場合には、ループ確率、非ループ確率ともに1/2になり、下位ビットn=2の場合には、ループ確率は1/4、非ループ確率3/4になり、下位ビットn=3の場合には、ループ確率は1/8、非ループ確率7/8になるから、下位ビットnの選択により繰り返しループ回数を変化させ、ネストから抜け出す速さを調節することができる。

【0072】また、カウンタの下位nビットの値が0でなければ、分岐先へ分岐することもでき、この場合も、下位ビットnの選択により、図11に示されているように、ループ確率、非ループ確率を変化させてネストから抜け出す速さを調節することができる。

【0073】これにより、DO文のネストを含む試験用命令列が出来上がり、DO文命令列およびDO文ネスト命令列による予測分岐試験が行われ得るようになる。

【0074】(実施の形態3)実施の形態3では、上記実施の形態1と2のものを組み合わせて一つの試験用命令列を生成する。これによりGOTO文、FOR文、DO文が混在した試験用命令列が生成される。

【0075】この実施の形態3の具体例を図12を参照して説明する。

【0076】(1)最初に、ネスト数を決定する。ここでは、DO文=3、FOR文=3とする。

【0077】(2)命令列生成過程で、ネスト数分の分岐先候補のポインタP(★)を無作為に抽出し、図10に示されているようなポインタ・テーブルPTに登録する。

【0078】(3)ネスト数分のポインタP(★)が設定された以降の命令列生成過程で、ネスト数分の無条件フォワード分岐命令UFB(●)を無作為に選択し、ネスト数分、ここでは3個の空き領域NR(□)を確保する。

【0079】(4)ネスト数分の空き領域NR(□)が確保された以降の命令列生成過程の任意の時点で、ネスト数分の無条件バック分岐命令UBB(◆)を生成する。各無条件バック分岐命令UBB(◆)の分岐先は、実施の形態1における場合と同等に、選択された空き領域NR(□)の先頭とする。

【0080】(5)選択された各空き領域NR(□)には分岐命令以外の任意の命令を埋め込み、空き領域NR(□)の最後には条件付フォワード分岐命令CFB(○)を埋め込む。この条件付フォワード分岐命令CF

B(○)の分岐先は分岐元の無条件バック分岐命令UB
B(◆)の直後とする。

【0081】(6)これより以降の命令列生成過程で、ネスト数分の条件付バック分岐命令CBB(◇)を選択する。条件付バック分岐命令CBB(◇)の分岐先は、実施の形態2における場合と同様に、ポインタ・テーブルPTに登録されているポインタP(★)によるネスト位置より選択設定される。

【0082】なお、条件付フォワード分岐命令CFB(○)、条件付バック分岐命令CBB(◇)の分岐条件値は、実施の形態1、2における場合と同様に、何れもタイマ受信命令により得られるカウンタ(タイマ)の値である。

【0083】これにより、DO文とFOR文のネスト混在の試験用命令列ができあがり、DO文とFOR文のネスト混在の命令列による予測分岐試験が行われ得るようになる。

【0084】

【発明の効果】以上の説明から理解される如く、請求項1に記載の情報処理装置の試験用命令列生成方法によれば、ランダムに生成可能な命令の一つとして、無限ループの問題を生じることなくバック分岐命令がランダム命令列に組み込まれるから、GOTO文やFOR文などのフロー制御文による実際の運用に近い分岐の検証試験が行われ得るようになり、ランダム命令生成の特質を活かした予測分岐試験を行うことができるようになる。

【0085】請求項2に記載の情報処理装置の試験用命令列生成方法では、分岐幅を可変設定できるから、ロング分岐を含む種々のフロー制御文による実際の運用に近い分岐の検証試験が行われ得るようになり、ランダム命令生成の特質を活かした予測分岐試験を行うことができるようになる。

【0086】請求項3に記載の情報処理装置の試験用命令列生成方法では、空き領域にフォワード分岐命令以外にランダム命令が挿入されることにより、バイ二ライン処理のタイミングが変化し、クリティカルな分岐テストが行われ得るようになる。

【0087】請求項4に記載の情報処理装置の試験用命令列生成方法では、ランダム分岐命令(GOTO文)が自動生成されるから、ランダム分岐命令の予測分岐試験が的確に行われ得るようになる。

【0088】請求項5に記載の情報処理装置の試験用命令列生成方法では、FOR文命令列あるいはFOR文ネスト命令列が自動生成されるから、FOR文命令列およびFOR文ネスト命令列による予測分岐試験が的確に行われ得るようになる。

【0089】請求項6に記載の情報処理装置の試験用命令列生成方法では、DO文命令列あるいはDO文ネスト命令列が自動生成されるから、DO文命令列およびDO文ネスト命令列による予測分岐試験が的確に行われ得る

ようになる。

【0090】請求項7に記載の情報処理装置の試験用命令列生成方法では、FOR文命令列、FOR文ネスト命令列、DO文命令列、DO文ネスト命令列における繰り返しループ回数が可変設定されるから、種々のフロー制御文による実際の運用に近い分岐の検証試験が行われ得るようになり、ランダム命令生成の特質を活かした予測分岐試験を行うことができるようになる。

【0091】請求項8に記載の情報処理装置の試験用命令列生成方法では、GOTO文、FOR文、DO文が混在した試験用命令列が生成されるから、種々のフロー制御文による実際の運用に近い分岐の検証試験が行われ得るようになり、ランダム命令生成の特質を活かした予測分岐試験を行うことができるようになる。

【図面の簡単な説明】

【図1】(a)～(c)は、この発明による情報処理装置の試験用命令列生成方法の実施の形態1の概要を示す説明図である。

【図2】(a)、(b)は、この発明による情報処理装置の試験用命令列生成方法で使用される空き領域ポインタ・テーブルを示す説明図である。

【図3】(a)は実施の形態1における無条件フォワード分岐命令の命令列の定義例を、(b)は無条件バック分岐命令の命令列の定義例をそれぞれ示す説明図である。

【図4】実施の形態1における空き領域ポインタ・テーブルを示す説明図である。

【図5】ランダム分岐命令(GOTO文)を含む試験用命令列の一例を示す説明図である。

【図6】(a)～(c)は、この発明による情報処理装置の試験用命令列生成方法の実施の形態1における命令生成過程を示す説明図である。

【図7】実施の形態1における無条件バック分岐命令の命令列の他の定義例を示す説明図である。

【図8】FOR文のループ確率表を示す図である。

【図9】(a)、(b)は、この発明による情報処理装置の試験用命令列生成方法の実施の形態2における命令生成過程を示す説明図である。

【図10】実施の形態2におけるポインタ・テーブルを示す説明図である。

【図11】DO文のループ確率表を示す図である。

【図12】実施の形態3におけるDO文とFOR文のネスト混在の試験用命令列を示す説明図である。

【図13】従来における試験用命令列の生成手順を示すフローチャートである。

【図14】(a)は命令生成域を、(b)、(c)は命令テーブルをそれぞれ示す説明図である。

【符号の説明】

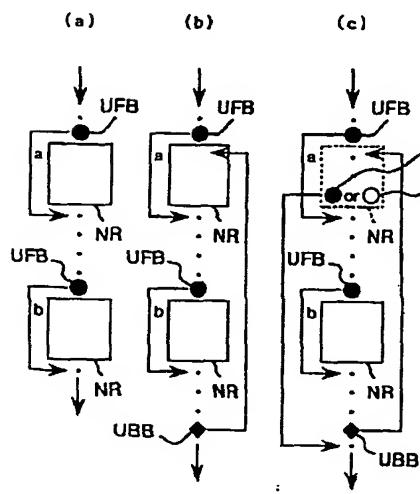
NR 空き領域(□)

UFB 無条件フォワード分岐命令()

CFB 条件付フォワード分岐命令 (○)
 UBB 無条件バック分岐命令 (◆)
 CBB 条件付バック分岐命令 (◇)

P 分岐先候補のポインタ (★)
 PT 空き領域ポインタ・テーブル

【図1】



【図2】

有効空間数2	
a	N
b	N

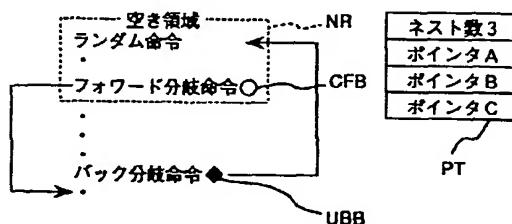
有効空間数1	
b	N

有効空間数2	
a	3
b	4
c	5

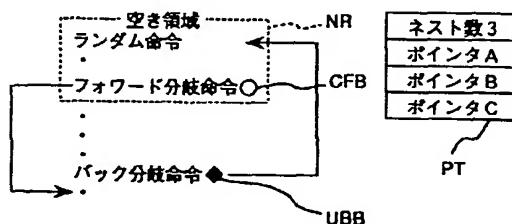
【図4】

有効空間数2	
b	4
c	5

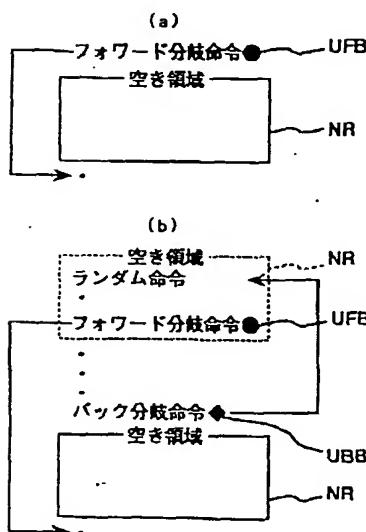
【図7】



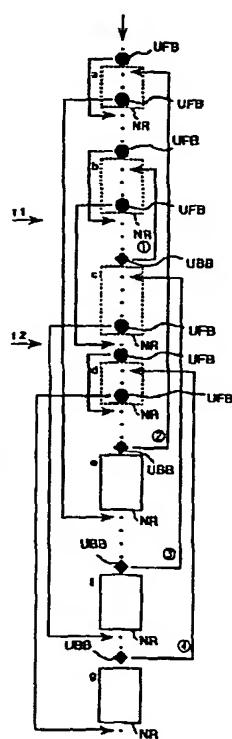
【図10】



【図3】

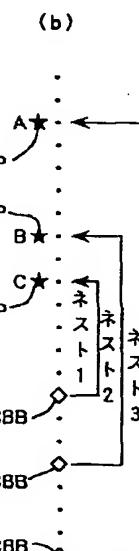


【図5】

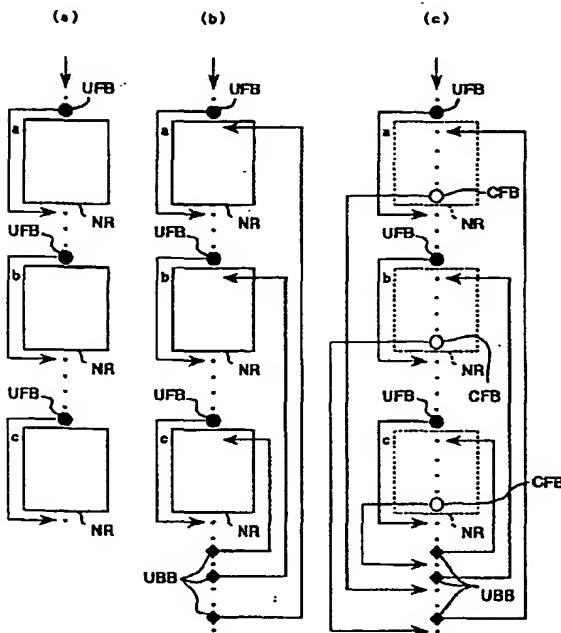


【図9】

- (a)
- (ランダム命令)
 - (ランダム命令)
- P A★
P B★
P C★
- 命令生成順序



【図6】



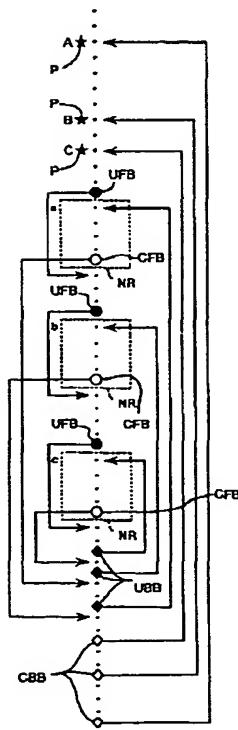
【図8】

下位ビット n	0で分岐の場合		0以外で分岐の場合	
	レーブ確率	非レーブ確率	レーブ確率	非レーブ確率
1	1/2	1/2	1/2	1/2
2	3/4	1/4	1/4	3/4
3	7/8	1/8	1/8	7/8

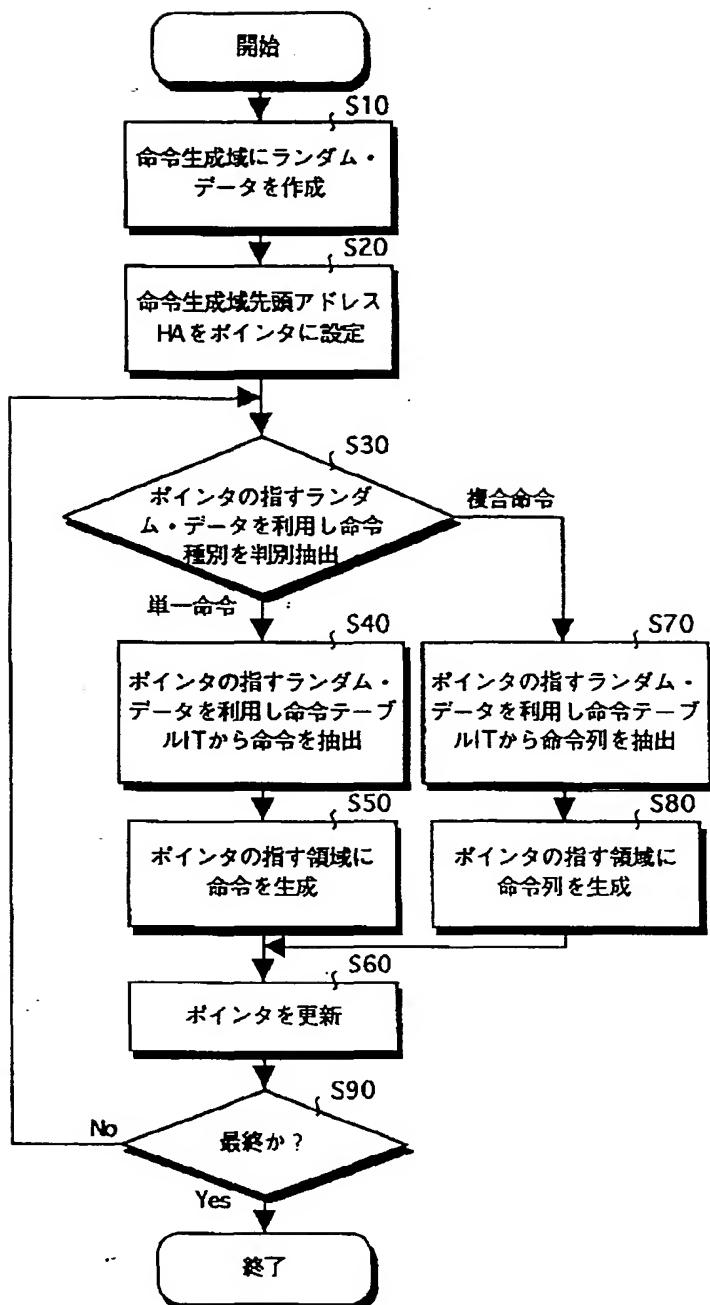
【図11】

下位ビット n	0で分岐の場合		0以外で分岐の場合	
	レーブ確率	非レーブ確率	レーブ確率	非レーブ確率
1	1/2	1/2	1/2	1/2
2	1/4	3/4	3/4	1/4
3	1/8	7/8	7/8	1/8

【図12】



【図13】



【図14】

